



Voice Over IP



Cambiando el aspecto de Mozilla



Nuevos Reportes de examenes Microsoft





PERIODICO DE NETWORKING

n°4





editorial

Seguimos creciendo: NEX ha despertado muchísimo interés entre los expertos de IT. Y por esto a partir de Febrero de 2004 estaremos con Ustedes mensualmente. Comenzando con una tirada mensual de 20.000 ejemplares entre suscriptores, centros de distribución y eventos de IT.

En éste número abordamos varias temáticas. Dos artículos describen los servidores WEB más populares: IIS 6.0 (Internet Information Server) de Microsoft y Apache creado por la comunidad Open Source. Este último articulo está complementado por uno sobre el Web browser Open Source (cliente) mozilla y cómo podemos modificarlo a las necesidades de nuestro enterprise.

Describimos el popular Microsoft Share Point Server. Un artículo de temática general le enseñará lo básico de Voz sobre IP (VoIP). Para quienes desarrollen Web pages en el artículo ASP vs. PHP se discute las ventajas de uno sobre el otro.

Los exámenes de certificación de Windows 2003 ya están disponibles y les mostramos preguntas ejemplo. Estas complementan preguntas de exámenes de la Certificación LPIC del Linux Professional Institute (LPI) y sobre el examen 70-228/229 de la carrera MCDBA (Microsoft Certified Data Base Administrator) usando SQL de MS. Completan los temas de certificación: sueldos en IT y los cambios introducidos por Microsoft al terminar de rendir un examen. Esperamos disfruten NEX4.

Invitamos a quienes deseen participar con notas, para ello deben contactarse a: articulos@nexweb.com.ar





Retire su ejemplar NEX en forma gratuita en Córdoba 657 piso 12° Capital Federal o solicítelo telefónicamente para su empresa al (011) 4312-7694
http: www.nexweb.com.ar

Staff

Año 2 Nº 4 2003

Director
Dr. Carlos Osvaldo Rodríguez

Propietarios
COR Technologies S.R.L.

Coordinador Editorial
Carlos Rodríguez Bontempi

Cordinación General
María Lujan Zito

Responsable de Contenidos
Dr. Carlos Osvaldo Rodríguez

Editor en Jefe
Raúl Kuzner

Redactores
Jorge Vazquez, Ing. Alejandra García
Germán Douek, José Gatti, Leonardo Costa
Raúl Kuzner, Emanuel Rincón y Marcelo Hernáez

Humor
Marcos Severi

Distribución
Lorena De Lillo

Diseño Web SIte
Emanuel A. Rincón

Diseño Gráfico
Marcos Ferrer

Publicidad
Ximena Antona
publicidad@nexweb.com.ar
4312-7694

Preimpresión e Impresión
Edigráfica s.a. Tel:4846236

Periódico de Networking
Registro de la propiedad Intelectual en trámite leg3038
Dirección: Córdoba 657 12°Capital Federal Tel:(011) 4312-7694 http: www.nexweb.com.ar

Tirada de esta edición: 20000 ejemplares

I-SEC Information Security

I-SEC INFORMATION SECURITY S.A., es la primera empresa Argentina que funciona como centro integral dedicado exclusivamente a la Seguridad Informática.

Cuenta con dos departamentos que se complementan en sus funciones:
Por un lado I-SEC INFORMATION SECURITY EDUCATION CENTER donde se han capacitando a mas de 1500 profesionales en el último año en diversas temáticas y especialidades que hacen a la Seguridad Informática (SI), brindando un enfoque que se rige por la norma ISO-17799.

I-SEC es un Centro de Estudios Internacional especializado en Seguridad de la Información y Auditoría de Sistemas que brinda diferentes alternativas de capacitación de acuerdo con su nivel de conocimientos técnicos, experiencias y posibilidades de tiempos y presupuestos. Principalmente sus entrenamientos están dirigidos a Profesionales de Tecnología de la Información, responsables de las distintas áreas de Sistemas, Tecnología, Auditoría, Seguridad Informática, Administración de Seguridad, Auditores y Consultores de *Information Technology* (IT, Tecnologías de la Información).

I-SEC tiene amplio reconocimiento en el mercado de la Tecnología, donde ya han participado más de 500 profesionales de compañías de primera línea en Latinoamérica: Buenos Aires, Santiago de Chile, Córdoba, Rosario y otras ciudades del Cono Sur.
Enfoque de Teoría según Normas Internacionales y Casos Prácticos de implementación. Examen y certificado final (opcional). Laboratorios técnicos directamente en los equipos informáticos. Los CV de los mejores promedios son presentados a las principales compañías de Argentina (opcional con autorización).

I-SEC tiene Instructores de Argentina y de USA, y brinda dentro de sus servicios: Especialización Anual en Seguridad de la Información y Auditoría de Sistemas ISO 17799; Certificación Internacional CISSP del ISC2; Information Security Technical Week; Conferencias Mensuales de Actualización Técnica; Conferencias de Certificación en Seguridad Norma ISO 17799; Entrenamiento Intensivo en Seguridad Norma ISO 17799.

Por otro lado I-SEC cuenta con su División de Consultoría y Auditoría especializada en Seguridad Informática, contando con capacidades técnicas y una amplia experiencia de más de 10 años en proyectos de Seguridad Informática:

- Conformada por un grupo de especialistas con experiencia local e internacional en la realización de trabajos en empresas de primera línea en Latinoamérica
- Asesora a Compañías según Norma ISO 17799 Seguridad de la Información
- Genera productos terminados con un enfoque detallado para una rápida implementación, utilizando herramientas automáticas de alto nivel basados en estándares internacionales y Normas ISO
- Cuenta con un equipo de ingeniería que está calificado especialmente en las últimas estrategias y tecnologías de seguridad.
- Realizan integraciones a los equipos de trabajo de otra consultoras y estudios

Servicios de Seguridad Informática

Outsourcing

- Búsqueda, selección, contratación, capacitación y reubicación de especialistas en las Compañías
- Outsourcing parcial o total de la Administración de la Seguridad perimetral / interna
- Monitoreo y mantenimiento permanente de la seguridad de las redes



SEGURIDAD DE LA INFORMACIÓN



Servicios de Seguridad Informática

Consultoría y Auditoría ISO 17799 Compliance

- Diseño e Implementación de Programas Integrales de Seguridad Informática basados en Norma ISO 17799
- Definición y puesta en marcha de la Función de Seguridad y Control Interno en la Compañía
- Desarrollo de Normas, Procedimientos y Estándares:
- Procesos administrativos con control interno
- Técnicos de Seguridad Informática
- Definición e implementación de medidas de seguridad para los sistemas de información:
  - sistemas operativos,
    - equipos de comunicaciones,
    - extranets, DMZ, firewalls,
  - correo electrónico,
    - ERP´s,
    - aplicaciones e-commerce,
    - encriptación,
    - bases de datos
    - Programas antifraudes informáticos
- Evaluación de la seguridad implementada:
  - intentos de penetración y técnicas de hacking,
    - diagnósticos integrales,
    - pruebas de auditoría
    - Auditoría Interna de procesos de negocios
- Auditorías de Seguridad para ISO 17799
- Soporte en Leyes y Regulaciones de Habeas Data, Software Legal
- Clasificación de la Información de la Compañía

Llega Info Security 2003

Al igual que en Estados Unidos y Europa, se realizará por primera vez en Argentina **Info Security 2003**, este encuentro de los profesionales de la Seguridad Informática, contara también con exposición de productos y servicios de vinculados a la Seguridad Informática. El Evento se desarrollará el 27 y 28 de noviembre en el Hotel Sheraton Libertador de la Ciudad de Buenos Aires (Av. Córdoba y Maipú).

Info Security 2003 será un espacio de oportunidades para contactarse con empresas, profesionales y directivos de IT (*Information Technology*-Tecnologías de la Información) y Seguridad Informática, y un lugar ideal para que los profesionales vinculados a estas áreas obtengan actualización técnica y perfeccionamiento.
Durante los dos días de la muestra se desarrollarán business meetings, conferencias, demos de soluciones y productos, seminarios intensivos y exposición de productos y servicios.

Info Security 2003 contará con un Auditorio Central, con capacidad para 500 personas, donde tendrán lugar dos seminarios de jornada completa, la primera jornada se dictará la Conferencia Ethical Hacking y durante la segunda jornada con un temario completo de la problemática y tendencias de la Seguridad Informática.

Adicionalmente se dispondrá de un Lab Center con capacidad para 100 personas, donde se desarrollarán conferencias sobre la actualidad de la Seguridad Informática. Por otra parte, la muestra contará con un Conference Room donde distintas empresas expondrán sus soluciones, productos y servicios, con la idea de generar contactos comerciales. Entre las compañías invitadas se encuentran consultoras nacionales e internacionales de servicios, comercializadoras y distribuidoras de productos de software y hardware, universidades con carreras de tecnología, representantes de Fuerzas de Seguridad, área de gobierno y diversas firmas del rubro.

Además, se realizará el primer ***I-SEC InfoSec Knowledge Test*** en Argentina entre los participantes a las conferencias, cuyos mejores promedios tendrán derecho a becas para el Programa 2004 de Especialización en Seguridad de la Información en el I-SEC.

Para acreditaciones e informes:
Info@i-sec.com.ar
www.i-sec.com.ar
ó a los Teléfonos
(011) 4342-7825 y 4343-0663



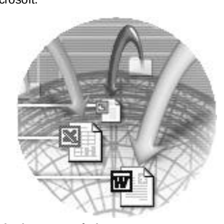
SharePoint Team Services y SharePoint Portal Server

En la familia de SharePoint se incluyen dos ofertas: Microsoft SharePoint Portal Server 2001 y SharePoint Team Services. Gracias a estas tecnologías pueden acceder a la información y organizarla, gestionar documentos y trabajar en equipo en proyectos. Todo ello en un entorno integrado con Microsoft Office y con un navegador con el que está familiarizado.

Los productos pueden utilizarse de forma independiente en la familia SharePoint o se pueden utilizar de forma combinada para ofrecer una solución completa que responda a las necesidades derivadas de la información compartida para organizaciones de cualquier magnitud.

¿Qué es SharePoint Team Services?
SharePoint Team Services es una solución para gestionar los sitios web de equipos que puede utilizar para crear áreas de trabajo según las necesidades para gestionar las actividades de los grupos y ofrecer resultados que se puedan compartir. Si quiere ver una descripción completa de las características y funciones de SharePoint Team Services, consulta la página con los Fast Facts de SharePoint.

¿Es lo mismo SharePoint Team Services que Office Web Server?
Sí. Office Web Server es el nombre clave; el nombre oficial es SharePoint™ Team Services de Microsoft.



¿Qué es SharePoint Portal Server?
Con SharePoint Portal Server 2001 se crear un sitio de un Portal Web que se puede utilizar par a compartir documentos y buscar información almacenada en cualquier lugar de la organización o la empresa, los sitios Web de SharePoint Team Services incluidos. Todo ello en una interfaz para portales de gran flexibilidad. Para más información, visite el sitio de SharePoint Portal Server.

¿Cuál es la diferencia entre SharePoint Team Services y SharePoint Portal Server?
En este diagrama se exponen las características de SharePoint Team Services y SharePoint Portal Server. Utilícelo para averiguar de qué forma puede integrar cada unas de las posibilidades de SharePoint en su empresa para sacarle el máximo beneficio.

¿Cómo puede adquirir SharePoint Team Services?
Hay diversas maneras:
*SharePoint Team Services está incluido en FrontPage 2002, que se ofrece de forma individual o como parte de la suite de Office XP para desarrolladores.
*También puede adquirir SharePoint Team Services como un servicio de suscripción a través de un WPP (Presence Providers).
*SharePoint Team Services es una característica integrada de la próxima generación de los sistemas operativos del Servidor de Windows.

¿Cuáles son los requisitos para obtener la licencia de SharePoint Team Services?
En comparación con la de cualquier otro producto o tecnología Microsoft, la licencia para SharePoint



Team Services es única:
*Para el servidor: Para poder implementar un sitio web para equipos de SharePoint Team Services necesita tener una licencia para FrontPage 2002. FrontPage 2002 se ofrece por separado como un producto y además se entrega con Office XP Developer.
*Para el cliente: No hay licencias de acceso de usuario (CALs) para SharePoint Team Services, independientemente de que se haya instalado en una red corporativa o esté disponible en un servicio alojado. Sin embargo, los administradores del servidor tienen que recurrir a los acuerdos de licencias para Microsoft Windows® y Microsoft SQL Server™ para los componentes de una solución que se base en SharePoint Team Services.

Notas adicionales:
*Para que un usuario pueda abrir un sitio Web basado en SharePoint Team Services con FrontPage 2002, su sistema debe tener una licencia para el acceso del usuario a FrontPage. Por lo general uno de cada nueve usuarios de SharePoint Team Services utilizan FrontPage para abrir el sitio Web del equipo.



*La venta de alojamiento está prohibida por el contrato de licencia para el usuario final (CLUF) para SharePoint Team Services, a no se que se trate de un Proveedor de presencia en la Web (WPP) para FrontPage y SharePoint Team Services. Los WPPs disponen de un apéndice que les permite vender el alojamiento de SharePoint Team Services.
Por ejemplo, una organización que facilita SharePoint Team Services a 1.000 usuarios tiene que disponer una licencia para FrontPage 2002 para el servidor y unas 100 licencias FrontPage para los usuarios que quieran personalizar los sitos web basados en SharePoint Team Services.

¿Qué necesito para poder utilizar SharePoint Team Services?
Cualquier cliente con un sistema operativo Windows puede utilizar los sitios Web creados con las características de SharePoint Team Services, siempre y cuando utilicen Microsoft Internet Explorer o Netscape Navigator 4.0 o posterior. También necesita tener un Servidor Windows 2000 o acceso al servidor a través de un proveedor de alojamiento de pago. Visite la página sobre Requisitos del sistema y averigüe si tiene todo lo que necesita para ejecutar SharePoint Team Services.

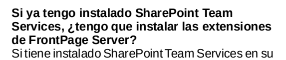
¿Necesito tener Office XP para poder utilizar SharePoint?
No. Se puede acceder a la mayoría de las



Tabla de características



Detalles	Team Services	Portal Server
Función principal	Información compartida ad hoc	Portal y búsqueda empresarial
Sitio Web	Sitios web de equipo	Sitios Web de Portal
Capacidades de búsqueda	Documentos en los sitios web y en los subwebs	Entre múltiples servidores y tipos de datos
Debates y avisos	Debates, Avisos y Sondeos	Discusiones y Notificaciones
Personalización	La versión 2002 de Microsoft FrontPage® basada en navegador y SDK	Web Parts y sDK
Gestión de documentos	Publicar	Control de Entrada y Salida, Control de versiones, Enrutamiento y Publicación
Aplicaciones para clientes	Navegador, Office XP, FrontPage 2002	Browser, Microsoft Windows Explorer, Office 2000 u Office XP
Niveles de seguridad según las responsabilidades	Responsabilidades personalizables: Administrador, Autor avanzado, colaborador y navegador	Administrador, Coordinador, Autor y Lector
Almacenamiento	Microsoft SQL Server™	Sistema de almacenamiento en Web
Licencias	Una licencia de FrontPage para el servidor; No hay licencias de acceso del usuario (CALs)	Licencia por servidor y CALs

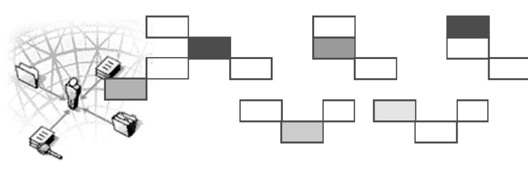


posibilidades de SharePoint, incluyendo a las más importantes, desde un buscador versión 4.0 o posterior. Sin embargo, sí es cierto que con Microsoft Office XP se disfruta de una mayor integración y más posibilidades, ver librerías de documentos en gráficos, un formato de fácil comprensión, información compartida sin problemas desde las aplicaciones Office a los sitios Web para los equipos y viceversa.

¿Necesito tener FrontPage 2002 para poder utilizar SharePoint?
No. Se pueden crear páginas nuevas, vistas específicas, administrar la seguridad y las autorizaciones con un navegador 4.0, también se pueden trabajar en el diseño de la página de inicio de los sitios Web de SharePoint Team Services. Con FrontPage 2002 puede agregar Temas, imágenes y bordes compartidos para darle un toque personal al diseño del sitio Web de su equipo. Además puede personalizar las listas y establecer las reglar del flujo de trabajo para las librerías de documentos.

Si ya tengo instalado SharePoint Team Services, ¿tengo que instalar las extensiones de FrontPage Server?
Si tiene instalado SharePoint Team Services en su servidor, no tiene que instalar las extensiones de FrontPage Server. SharePoint Team Services ofrecen un set superior de las posibilidades de las extensiones de FrontPage Server.

Preguntas y respuestas de administradores de sistemas
¿SharePoint Team Services es una tecnología de servidor o de cliente?
Ninguna de las dos. SharePoint Team Services es una tecnología de Windows® con la que los clientes pueden ejecutar una versión estándar 4.0 o posterior para colaborar y compartir información o documentos utilizando un sitio Web. SharePoint Team Services se instala en un servidor que ejecute Windows 2000 Server o Windows Advanced Server y Internet Information Services (IIS) 5.0 al que se accede en el cliente que utilizar Internet Explorer o Netscape Navigator 4.0 y posterior.



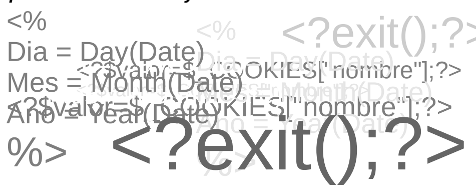
¿En qué plataforma se ejecuta SharePoint Team Services?
SharePoint Team Services se ejecuta en plataforma Windows 2000 (con Service Pack 1) con IIS 5.0. Por defecto, SharePoint Team Services instala Microsoft Data Engine (MSDE) para ofrecer soporte a los requisitos de las bases de datos en entornos dedicados y utiliza SQL Server para cuando hay mucho tráfico o entornos de alojamiento compartido. SharePoint Team Services es además una característica integrada en las versiones futuras de algunos de los sistemas operativos de Windows Server.

¿Hasta que punto es escalable SharePoint Team Services?
Con un único servidor ejecutando SharePoint Team Services con MSDE pueden ofrecer unos resultados más que buenos a unos niveles moderados de utilización, ofreciendo soporte a cientos de sitios y por lo tanto a miles de usuarios. Sin embargo, si sustituye MSDE por SQL Server 2000, puede ampliar de manera importante la capacidad, especialmente si las tareas de las bases de datos se descargan en un servidor de bases de datos específico. Por ejemplo, en Microsoft, cuatro servidores con procesadores cuádruples de 800 MHz equipados con 3 GB de memoria RAM han podido ofrecer soporte a 15.000 sitios Web para equipos con 50.000 usuarios.

Por Marcelo Hernáez
Instructor MS Office (MCT)

Comparación entre el lenguaje de programación C# y Java.

Dare Obasanjo ha escrito una muy interesante comparación entre el lenguaje de programación C# (C sharp) y Java. Aquí extraemos algunas ideas de su artículo a modo de tener un pantallazo de tal comparación. Quien esté interesado en conocer la comparación en detalle puede referirse a *http://www.25hoursaday.com/events#events*



Dare nos dice que:
El lenguaje C# es un lenguaje de programación object-oriented (orientado a objetos) que apunta a permitir a los programadores construir muy rápidamente un amplio rango de aplicaciones para la plataforma .NET de Microsoft. La meta de C# y la plataforma .NET es la de acortar el tiempo de desarrollo liberando al desarrollador de preocuparse sobre varios temas de bajo nivel tales como administración de la memoria, "*type safety issues*", construcción de librerías de bajo nivel, chequeo de límites en los arreglos (array bounds checking), etc., permitiendo a los desarrolladores usar su tiempo y energía trabajando sobre la aplicación y la lógica de negocio asociada. La oración anterior podría representar una breve descripción del lenguaje Java y su plataforma si las palabras "C#" y "plataforma .NET" se reemplazaran por las palabras "Java" y "plataforma Java" respectivamente.

En su artículo nos da una visión general de las similitudes y diferencias entre características de los lenguajes y librerías del lenguaje C# y Java basándose en su experiencia en el uso de ambos, Da ejemplos de "*code snippets*" (pequeños programas) que fueron testeados en Microsoft .NET Framework para snippets en C# y Java 2, Standard Edition (J2SE) para el caso de snippets en Java.

Como conclusión detalla que:

La mayoría de los desarrolladores, especialmente aquellos con una base en C o C++, convendrían probablemente en que características como la sobrecarga de operadores, los punteros, directivas al preprocesador, delegates y la limpieza determinística de objetos hacen que C# sea más expresivo que Java en un número de casos. De forma similar, los desarrolladores de Java que aprenden C# serán sorprendidos agradablemente por las características que faltan en Java una vez que las utilizan en C#, por ejemplo: boxing, las enumeraciones y pass por referencia. Por otra parte la carencia de excepciones comprobadas, de clases internas, de la portabilidad de la plataforma cruzada o del hecho de que una clase no sea la unidad más pequeña de la distribución de código hacen que la opción de C# sobre Java no sea clara para elegir más características de lenguaje sin tener que hacer un compromiso.

Es mi opinión que ambos lenguajes son tan similares que podrían ser hechos para reflejarse mutuamente sin gran esfuerzo si fuera requerido por parte de usuarios de cualquier base. En este caso, C# tendría la ventaja sobre Java en que C# tiene menos cosas que pedir prestadas de Java de las que Java tendría que pedir prestadas de C#. Sin embargo, el valor verdadero de un lenguaje de programación que se piensa para el uso fuera del ámbito académico es cuan rápidamente se desarrolla para adaptarse al entorno tecnológico cambiante y qué clase de comunidad rodea al lenguaje.

Algunos lenguajes de programación se asemejan con el francés de *Les Immortels de la Académie Française* en Francia. *Les Immortels* se encarga de dictaminar qué constituye la lengua francesa oficial pero ha sido lento al adaptarse a la era de la información, por lo tanto sus edictos, en lo que se refiere a las versiones francesas apropiadas de nuevas palabras, especialmente las relacionadas con la tecnología, no son especialmente tomadas en cuenta, puesto que entran en conflicto con lo que normalmente utiliza el público francés y con las palabras no sancionadas que ya se han incorporado al léxico. C++ es un ejemplo de un lenguaje que ha experimentado un proceso que se asemeja mucho a la lengua francesa bajo *Les Immortels de la Académie Française* mientras que Java bajo *Sun Microsystems* puede ser considerado un lenguaje que se ha ido desarrollando con el tiempo , al igual que la lengua inglesa lo ha hecho.

Entonces la pregunta es: ¿Cuál de estos lenguajes pareciera que evolucionará con el tiempo y será el más adaptable a situaciones a medida que se presentan? Hasta ahora, *Sun* ha hecho un gran trabajo con Java aunque haya una carencia de soporte en las versiones y el hecho de que no exista un *framework* que permita la extensión del lenguaje construido en la plataforma hacen que la evolución sea drásticamente difícil. C#, con su soporte para versiones vía *framework .NET* y la existencia de atributos que se pueden utilizar para extender las características del lenguaje, parecen indicar que a la larga sería el lenguaje más adaptable. Solamente el tiempo nos dirá, sin embargo, si esta predicción es exacta.

Quien se interese en saber más sobre *Dare Obasanjo* ver su página web: *http://www.25hoursaday.com/*

Recursos

*Eckel, Bruce. Thinking In Java. Prentice Hall, 2000.
*Gunnerson, Eric. A Programmer's Introduction to C#. Apress, 2001.
*Sun Microsystems. The Java™ Tutorial. *http://java.sun.com/docs/books/tutorial/*
*Microsoft Corporation. .NET Framework SDK Documentation. *http://msdn.microsoft.com/library/default.asp?url=/library/en-us/nfstart/html/sdkstart.asp*



IIS 6.0

Las mejoras en Windows 2003 no fueron pocas si se suman las introducidas en cada uno de los servicios del mismo, entre ellos cabe destacar el conjunto de servicios IIS 6.0. Muchos de estos cambios llegan para resolver aspectos de la arquitectura y detalles técnicos de IIS 5.0.

Pasaremos a describir estas nuevas características del producto Internet Information Services clasificándolas en 3 grandes áreas:

Seguridad
No es sorpresa escuchar a diario de los ataques que sufren los servidores expuestos a Internet (y hoy día hasta las PCs hogareñas!!) derivados de exposiciones a ataques que dan lugar los programas que residen en los mismos que brindan los servicios de Internet. Por supuesto que esto no es exclusivo de ningún software en particular. Por eso uno de los lemas de aquel que quiere seguridad es : "funcionalidad no es igual a seguridad". A diferencia de Windows 2000 e IIS5 en Windows 2003 IIS6 NO viene instalado por defecto. Y, cuando se instala IIS6 se hace con lo que seria un Lock-down Status. Esto quiere decir que por defecto Ud. tendrá una versión "liviana" (solo para contenido estático) del producto y si quiere habilitar funciones avanzadas para trabajar con contenido dinámico por ejemplo (aquellos que requieren código ejecutándose en el servidor) tendrán que ser habilitadas expresamente por el administrador. Esto resulta muy útil. En la versión 5 ya existía una utilidad similar, la IIS lockdown Utility.
Veamos los cambios más significativos respecto seguridad:

Advanced Digest Authentication
Una extensión de la ya provista Digest Authentication que usa el algoritmo MD5 para la encriptación de credenciales (contraseñas) cuando se transmiten vía HTTP, una de las ventajas de esta extensión es que las contraseñas, a diferencia de la versión anterior, es que ya no deben ser almacenadas en Active Directory mediante algoritmos de encriptación reversible sino que la mismas se almacenen directamente en forma de hash, de encriptación no reversible, mediante también el algoritmo MD5

Server-Gated Cryptography
Una versión más poderosa de ese sistema de encriptación similar a SSL/TLS, ya introducido con IIS4 pero ahora con soporte de encriptación de 128 bits.
Aplicaciones del sector financiero (sistemas bancarios o financieros) son los usos comunes de SGC.

Selectable Cryptographic Service Provider (CSP).
La traducción a esto seria algo así como Proveedor de Criptografía Seleccionable, bien para eso hay que entender un poco que es un CSP: Un CSP es el mecanismo que implementamos (ya sea de MSFT o de terceros) para encriptar, almacenar y manipular claves de encriptación, típicamente asimétricas (publicas y privadas) en los diferentes medios de almacenamiento disponibles para alojar estas, ejemplos de estos incluyen la Registry de Windows, Smart Cards, Token, etc.,
Entonces esto indica que aparte de contar con las provistas por MSFT, que ya están optimizadas, uno puede decidir implementar otras CSP´s con el fin de cambiar de medio de almacenamiento (Ej.: hardware propietario), performance o seguridad.

Identidad configurable de un Worker Process
El entorno donde corren los Web sites o aplicaciones que corren como procesos aislados en IIS 6.0 se llama Worker Process (de esto hablaremos luego), bueno estos pueden ser configuradas para correr bajo una cuenta de mucho menor privilegio que la corre le mismo proceso de IIS 6.0.

Nuevo Framework de Autorización
Autorización es el proceso de confirmar el acceso de un usuario a un recurso, a diferencia de la Autenticación que es el proceso de identificar quien es el usuario. Después de la Autenticación se realiza típicamente la Autorización.
Hay dos tipos de opciones de autorización para ASP.NET disponibles para IIS 6.0:
*File Autorización: vía la autenticación integrada de Windows primero luego se verifica la ACL (lista de Control de Acceso) del archivo ASP.NET(.asmx) para dar acceso a la aplicación o al Web site.
*URL Authorization: esto permitiría almacenar la información de autorización en un archivo XML mediante el uso tags.

Confiabilidad
Si hablamos de este aspecto habría que destacar entonces el Worker Process Model, que como hemos mencionado anteriormente, también afecta la seguridad (Identidad configurable de un Worker Process). IIS tiene la cualidad de separar todo el código del usuario del servicio WWW. Ya era el caso de IIS 5.0 que fue diseñado para poseer un solo proceso, el Inetinfo.exe, que funciona como proceso Web Server principal, que es el que puede derivar solicitudes a una o mas aplicaciones fuera de proceso (out-of-process) implementadas bajo un dllhost.exe (grafico #1).

Los procesos que corren en modo kernel del sistema operativo o también llamado modo protegido, como HTTP.SYS, no se ven afectados por procesos que corren en modo usuario.

En esta arquitectura, Inetinfo.exe es el proceso maestro a través del cual deben pasar todas las solicitudes sin importar el modo de aislamiento de proceso que se configure, ya sea el In-Process, Out-of-process o Pooled. Se pueden en IIS 5, configurar aplicaciones de alta prioridad para que corran como aisladas (Isolated), esto crea otra instancia de dllhost.exe, pero esta aislamiento fuera de proceso (out-of-process), mientras protege mejor al Web Server, es lenta en términos de rendimiento.
Una buena opción equilibrada en términos rendimiento/aislamiento es el modo Pooled, pero había un solo pool (conjunto mancomunado) de aplicaciones por servidor y todas las aplicaciones que usen este modo usarían el mismo pool.

La versión 6.0 de IIS ha sido totalmente rediseñada con una nueva arquitectura de procesamiento de solicitudes que posee los beneficios de una alta aislamiento, sin incurrir en los costos que implica usar mecanismos out-of process, así como mayores niveles de escalabilidad y confiabilidad.
Podríamos decir que IIS fue rediseñado en dos grandes partes:
*HTTP.SYS: encargado de escuchar las solicitudes, atenderlas y encolarlas pertinentemente a la cola de un determinado pool de aplicaciones, conectar a base de de datos las paginas ASP.NET, Cachear paginas Web, manejo de ancho de banda (Bandwith Throttling) y

IIS 5.0 Architecture

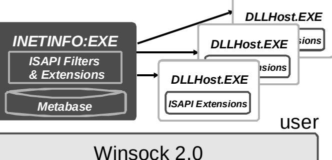


Gráfico #1

IIS tiene la cualidad de separar todo el código del usuario del servicio WWW

Preguntas Examen 70-290

Administración y Mantenimiento de Windows Server 2003

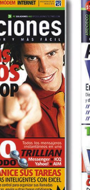
En Windows Server 2003, en IIS 6.0, editar directamente la metabase XML está:

A- Soportada
B- No soportada
C- Soportada, instalando el Service Pack
D- Ninguna de estas es correctas

Rta: A











generación de registros en archivos de texto para auditar IIS. Los procesos que corren en modo kernel del sistema operativo o también llamado modo protegido, como HTTP.SYS, no se ven afectados por procesos que corren en modo usuario.
*WAS: el administrador de procesos (Worker Process) y configuración. (Ver grafico #2).

IIS 6.0 tiene dos modos de operación.
-*Worker Process Isolation Mode* (por defecto): que corre todo el código de las aplicaciones en un entorno aislado, pero sin las consecuencias de perdida de rendimiento que sufrían las anteriores versiones de IIS, ahora las solicitudes son encaminadas a través del driver que trabaja en modo kenel (HTTP.SYS) al correcto pool de aplicaciones. Procesos en modo usuario sirviendo a un determinado pool de aplicaciones toman las solicitudes directamente de HTTP.SYS y eliminan los innecesarios saltos de proceso que suscitan cuando hay que enviar la solicitud a una DLL fuera de proceso (out-of process) y volver.
Todo el procesamiento de la aplicación Web, incluyendo la carga de los ISAPI filters y extensiones , así como la autenticación y autorización, es hecho por una nueva DLL del servicio WWW, esta dll se carga dentro de uno o mas procesos que la hospedan llamados Worker Process. El ejecutable de los Worker Process se llama W3WP.exe. Los Worker Process se ejecutan en modo usuario, obtienen la solicitud de la cola y lo encaminan al pool de aplicaciones. Esto lleva a niveles de aislamiento superior a las aplicaciones pero sin impacto en términos de performance.
-*IIS 5 compability isolation mode*: el equivalente al modo IIS 5.

Health Detection
Esta funcionalidad vendría a ser una "verificación de salud" de los Worker Process de IIS 6.0. En este caso el proceso inetinfo.exe verifica la disponibilidad de cada Worker Process periódicamente. Se puede configurar el intervalo para mantener un "heart beat" (latido) para así verificar la vida del Worker Process (similar a un ping).



IIS 6.0 Architecture
user
Web Admin Service
Worker Process W3 Core
Web app
HTTP.SYS
kernel

Gráfico #2

Funcionalidad
Estamos acostumbrados a hablar de ASP como entorno de programación cuando nos referimos a IIS, bueno IIS 6.0 utiliza ASP.NET que por cierto es mucho mas que la sola adición del texto .NET, por no decir que es un mundo nuevo, para darles algunas datos a aquellos que usaban el viejo ASP:
*ASP.NET esta basado en el framework Microsoft.NET: al igual que ASP se puede escribir código en VBScript, Jscript, etc. Se puede codificar ASP.NET en cualquier lenguaje compatible con el Common Language Runtime (CLR) de Microsoft, o sea, en la actualidad: Visual C#, VB.NET, Jscript.NET y J# sin contar todos los que están en desarrollo para el CLR.
*Se puede codificar en uno o más lenguajes en la misma página ASP.NET.
*El código ASP.NET se compila de antemano y no se interpreta línea a línea como ASP.
*ASP.NET soporta tres niveles de cache. Se puede cachear paginas completas, partes seleccionadas de una pagina (fragment caching) o mediante la Caching API, la cual permite a los desarrolladores tener un control muy preciso sobre como manejar el cache y así incrementar la performance.
*Archivos de registro (log files) Multilinguales gracias al soporte de Unicode Transformation Format-8 (UTF-8)

Metabase XML
La Metabase es donde IIS almacena la configuración, que en esta versión a diferencia de las anteriores (binarias) se guarda en el formato Extensible Markup Language (XML).
Características:
*Se puede editar con cualquier editor de texto plano, aun con IIS 6.0 en ejecución (edit while running).
*No se necesita reiniciar IIS para reflejar los cambios salvo que lo que se edite sea el archivo del esquema (schema file).
*Se redujo considerablemente los tiempos de inicio y apagado de IIS
La Metabase esta compuesta por 2 archivos
*Metabase.xml: un documento XML que contiene los valores de configuración para el servidor IIS, como puede ser la configuración de Web sites y Virtual directories.
*MBSchema.xml: Un documento XML que aloja el esquema (schema) de la Metabase XML, Este actúa como validador de los valores que se introducen en el archivo Metabase.xml.

Estos son los cambios significativos introducidos en esta nueva versión de IIS, por supuesto hay más, será solo cuestión de investigarlos a medida que se programe o se cree contenido Web.

Germán Douek
MCSE/MCT
Fuente: Devx.com (Chris Peiris), Sam Gentile, Capricorn Consulting (Stuart Dickerson)

Se puede codificar ASP.NET en cualquier lenguaje compatible con el Common Language Runtime (CLR) de Microsoft.

VERANO 2004







Technologies

Mucho más que un centro de Capacitación

FAST TRACKS

>> Microsoft BOOT CAMP 04

Incluye UPGRADE Windows 2003 Server System

	Horas de Capacitación
MCSA BOOT CAMP 80 hs => Promo : **1490 $ + IVA** (Incluye 120 Cor Cheks)*	80 Hs **+ 16 hs Upgrade MCSA W2003**
MCSE BOOT CAMP 120 hs => Promo : **2599 $ + IVA** (Incluye 250 Cor Cheks)*	120 Hs **+ 40 hs Upgrade MCSA W2003**

9 Enero 04 >> L a V de 18:30 a 22:30
9 Enero 04 >> L a V de 9:00 a 18:00
5 Febrero 04 >> L a V de 18:30 a 22:30
5 Febrero 04 >> L a V de 9:00 a 18:00
} **Inicio**

Microsoft CERTIFIED Systems Administrator — Microsoft CERTIFIED Systems Engineer

Microsoft Windows Server 2003

Más Información en http://VERANO.CORTECH.COM.AR ó http://www.cortech.com.ar/VERANO *Promoción limitada hasta agotar 120 Unidades por carrera o hasta el 31 Diciembre 2003.

>> Linux Verano 04

Incluye Taller LINUX: Aplicaciones y Distribuciones

	Horas de Capacitación
Linux Completa 45 hs => Promo : **440 $ + IVA** (Incluye 200 Cor Cheks)*	45 Hs **+ 12 hs Taller Linux: Aplic. Y Dist.**
Linux Expert 69 hs => Promo : **800 $ + IVA** (Incluye 400 Cor Cheks)*	69 Hs **+ 12 hs Taller Linux: Aplic. Y Dist.**



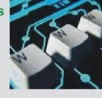
7 Enero 04 >> L a V de 18:30 a 21:30
9 Enero 04 >> L a V de 10:00 a 17:00
11 Febrero 04 >> L a V de 18:30 a 21:30
12 Febrero 04 >> L a V de 10:00 a 17:00
} **Inicio**

Más Información en http://VERANO.CORTECH.COM.AR ó http://www.cortech.com.ar/VERANO *Promoción limitada hasta agotar 120 Unidades por carrera o hasta el 31 Diciembre 2003.

>> WEB Design Verano 04

Incluye Taller WEB: Scripts y Hints

	Horas de Capacitación
WEB Design Completa 60 hs => Promo : **520 $ + IVA** (Incluye 260 Cor Cheks)*	60 Hs **+ 12 hs Taller WEB: Scr. y Hints**
WEB Design Expert 100 hs => Promo : **800 $ + IVA** (Incluye 400 Cor Cheks)*	100 Hs **+ 12 hs Taller WEB: Scr. y Hints**



7 Enero 04 >> L a V de 18:30 a 21:30
7 Enero 04 >> L a V de 10:00 a 17:00
12 Febrero 04 >> L a V de 18:30 a 21:30
12 Febrero 04 >> L a V de 10:00 a 17:00
} **Inicio**

Más Información en http://VERANO.CORTECH.COM.AR ó http://www.cortech.com.ar/VERANO *Promoción limitada hasta agotar 120 Unidades por carrera o hasta el 31 Diciembre 2003.

Preparandote para las correspondientes

Certificaciones Internacionales

Microsoft, Linux Professional Institute y Macromedia.











WWW.CORTECH.COM.AR

Voice Over IP (Voz sobre IP)

Una de las tecnologías emergentes hoy es “Voz Sobre IP” (Voice Over IP-VoIP). Aquí la analizamos en particular para dispositivos móviles.

Aclaremos un primer punto: VoIP es ilegal en algunos países. Aquellos, donde las telecomunicaciones nacionales son un monopolio. Por ejemplo USA, VoIP es totalmente legal.

¿Qué es VoIP?
El que comienza habla sobre un micrófono que está conectado a la placa de sonido de la PC. El micrófono detecta las ondas sonoras de la voz y transforma éstas en una señal analógica que la tarjeta de sonido sabe procesar. Se digitaliza una señal. Un proceso “codec” (corder-decoder) se utiliza entonces para comprimir la señal digital que viajará por la red local y luego Internet. La PC del que recibe realiza el proceso inverso y “escucha” el mensaje. Cuando este proceso es realizado simultáneamente se llama: conexión full-duplex.
Normalmente los paquetes de Internet viajan sobre conexiones de anchos de banda más grandes que una línea telefónica analógica standard.
Existe sin embargo, una diferencia grande entre ambas tecnologías: las líneas de transmisión telefónicas usan PSTN (Public Switched Telephone Network) de modo que la transmisión ocurre como un proceso de datos que fluyen en forma continua. Los datos por “paquetes” que se transmiten por Internet no lo hacen en forma continua, ni en orden (algunos paquetes hasta pueden perderse). Tal proceso con discontinuidades, falta de orden de paquetes y aún paquetes perdidos hace que escribir software codec sea difícil. Sin embargo se puede y ha sido hecho.

¿Entonces, porqué y cómo hacer VoIP?
La mayoría de las PC bajo Windows tienen un producto llamado NetMeeting (Start/Programs/Accesories/Communications/Net Meeting). Si corremos 2 PCs en una LAN, cualquiera de ellas puede llamar a la otra por su número IP. Si tengo micrófonos y auriculares (o parlantes) la calidad es excelente. La pregunta es para qué hablar por la PC cuando hay un teléfono pegado a cada una de ellas. Qué pasa si los teléfonos están en diferentes ciudades?. La llamada es larga distancia nacional o internacional. ¡VoIP es totalmente gratis!

Gateways de VoIP
Poder realizar “llamados” entre computadoras es interesante, pero para que esta tecnología nos sea realmente útil deberá poder hacer “llamadas” a teléfonos convencionales.
Para ello debemos acceder a un “gateway” VoIP. Este es usualmente un router especializado que tiene funcionalidad sobre IP y PSTN. Estas gateways usan típicamente protocolo ITV (Internacional Telecomunications Union) H.323 (http:/www.openh323.org) o el protocolo “Session Initation” de la (SIP) Internet Enginnering Task Force (ITETF). (SIP-http://www1.cs.columbia.edu/sip).
Ambos protocolos le permiten cambiar entre data-stream multi mediales en tiempo real y paquetes digitales de datos.
Para sucursales o pequeñas oficinas/hogar (small office/home SOHO) uno puede utilizar gateways públicas (empresas que proveen un número IP y hacen la conexión a la red telefónica.) Por ejemplo en USA un gateway puede ser “iconnect here” de deltathree (Ver www.deltathree.com.) No es gratis pero es posible realizar comunicaciones larga distancia a una fracción del gasto telefónico clásico.
Además esta empresa le provee uso de llamados de teléfono a teléfono (accediendo a un número telefónico especial y entrando número de usuario y PIN) a un costo reducido.

¿Qué software se usa para cada tipo de servicio?
En el caso de “iconnecthere” se usa PCphone que corre bajo Windows en desktops, notebooks y TabletPC. Con este software podemos “llamar” otra computadora por su IP del mismo modo que NetMeeting. También lo puede usar para hacer una llamada convencional. La aplicación abre una sesión SIP con el gateway que se encarga de establecer la llamada.
Todo lo descrito anteriormente funciona igualmente bien sobre redes wireless. Usando el software apropiado se puede hasta usar una pocket PC para hacer VoIP.
Se viene hablando hace tiempo, de los dispositivos convergentes (convergent devices). Aquellos que fusionan las funciones de un teléfono celular y un PDA. Aunque estos productos existen son: extremadamente caros, ligados a un proveedor de telefonía celular y con pocas funcionalidades en el PDA. Con aplicaciones VoIP uno puede transformar una Pocket PC, Tablet PC o notebook con el hardware de audio apropiado en un dispositivo convergente siempre y cuando estemos el alcance de un AP (acces Point) Wireless (Para entender el concepto de AP lea NEX 2: “El ABC de Redes inalámbricas”)



¿Qué pasa con la performance?
Los problemas pueden surgir de los firewalls de nuestra red y gateways que usan NAT (network Address Translation).
Dependiendo de los protocolos que use nuestra configuración VoIP, ciertos puertos en firewalls deben estar abiertos.
Un software como PC Phone, nos da un Test que determina si una conexión a Internet con VoIP funcionará.
Para medir la performance debemos “pingear” el IP de las máquinas de interes o del gateway PSTN. Si el tiempo de la respuesta es mayor que 250 milisegundos (ms) VoIP no funcionará aceptablemente.
Las corporaciones y empresas se cuestionarán qué ancho de banda necesitarán para poder realizar VoIP. Si usamos la herramienta Network Monitor de XP sobre una conexión inalámbrica (por ejemplo) veríamos que no usa más del 1% del bandwidth de una típica conexión 802.11b (cerca de 100kbps).
Es decir un AP (access point) podría soportar hasta 100 “llamadas” VoIP simultáneas.
No sólo uno puede enviar las llamadas sobre el gateway de VoIP, pero también recibir llamadas en la PC. Se ahorra con el VoIP y es posible implementarlo siempre. Todo dependerá de cada problema particular. Acostumbrarse a usar la PC es un factor. El segundo va a depender del ancho de la banda disponible y lo que se llama latency.
Como ya mencionamos la demora que introduce la “latency de la red” se hace notable para el caso de transmisión de voz cuando el tiempo de ida y vuelta de los paquetes es mayor de 250ms. Por eso debemos asegurarnos un one way latency menor que 150 ms. La mayoría de las personas pueden soporta una latency no mayor de 500 ms. Los satélites usualmente reportan latencies entre 800 y 1000 ms. Recordemos que aunque tal latency no es problema para transferencia de datos, si lo es para tráfico de voz.

El futuro
Muchos creen ver en VoIP una tecnología revolucionaria. Especialmente, combinada con el mundo de redes wireless. Veremos.
El ahorro pareciera ser uno de los motivos a favor de adoptar este tipo de tecnología. Pero este factor dependerá de donde y en que momento se trata de implementar. Por ejemplo en Argentina en el 2003 habiendo pasado la devaluación no hay ahorro en implementar VoIP. Los enlaces de datos y los equipos necesarios para realizar esto se incrementaron casi en un factor tres frente a un pequeño aumento de las comunicaciones telefónicas.

PDA (Personal Digital Assistant)
Un dispositivo de mano, que combina funcionalidades de computación, telefonía/fax, Internet, y redes.
Un PDA típico, funciona como teléfono celular, enviador de fax, Web browser y organizador personal. A diferencia de una “computadora portátil” la mayor parte de los PDA comenzaron usando una lapicera más que un teclado. Para poder entrar la información. Esto significo que además incorporaron reconocimiento de escritura a mano. Algunos PDA también pueden entender comandos por voz Hoy se las ofrece en versiones con lapicera y/o teclado.



Para encontrar contenido relacionado: www.mobile-and-wireless.com y www.recursosvoip.com (en español)

Preguntas para Examen 70-270

Instalación, Configuración y Administración de Windows XP Professional

Usted es el soporte técnico de su empresa. Todos los empleados usan PCs con Windows XP Professional. Un usuario llamado Miguel es miembro del grupo Ventas. El reporta que no puede salvar una planilla de cálculos del reporte de ventas a otra PC. Usted examina las propiedades de la carpeta compartida que tiene los siguientes permisos:

Tipo de Permiso	Permiso Concedido	Usuario o Grupo
Carpeta Compartida	Control Total	Todos
NTFS	Leer y Ejecutar	Ventas
NTFS	Modificar	Gerentes
NTFS	Control Total	Administradores

¿Qué debería hacer usted para asegurarse de que Miguel pueda guardar la planilla de cálculo en la carpeta compartida?

A- Dar a Miguel el permiso de Modificar de Recurso Compartido
B- Dar al Grupo Ventas el permiso Modificar de NTFS
C- Dar a Miguel el permiso Control Total de Recurso Compartido
D- Dar al Grupo Ventas el permiso Crear Archivos

Rta: B

Servidor Web Apache

El Webserver Apache es uno de los éxitos de los desarrollos de Open Source. Es estable, rápido, extremadamente flexible, y está disponible para un amplio rango de sistemas operativos (Windows, Linux, Unix). Su competidor es el IIS (Internet Information Server) de Microsoft. Más del 60 % de los sitios web en la actualidad corren sobre Apache. Pero ¿es difícil de configurar y usar?

La respuesta es que es sumamente sencillo hacerlo realizar cosas simples a partir de una instalación básica por defecto. Por ejemplo un sito Web estático. Luego, uno puede aprender más y agregar gradualmente nuevas prestaciones.
Aquí, veremos como instalar Apache con una configuración default (por defecto). En un próximo artículo se verán las posibilidades que ofrecen algunos de los módulos fundamentales de Apache y cómo configurarlos. Primero aprendamos sobre la administración y configuración del server. No nos detendremos mucho en cómo diseñar un contenido Web. El siguiente paso será aprender cómo levantar un Website que soporte virtual hosting, password-protection, logging, server-side scripting, y muchas otras funcionalidades.

3 métodos de instalación

Hay varias formas de instalar Apache en su sistema Linux.
* **Método 1**: puede incluirlo cuando instale su distribución (Suse, Mandrake, Red Hat) de Linux.
* **Método 2**: puede instalarlo desde un RPM binario.
* **Método 3**: puede instalarlo desde la fuente (códigos fuente en C que necesitaremos compilar), bajada desde un archivo de internet.

Lo primero que debería hacer es usar el comando RPM para hacer una "prueba de instalación" activándolo con la opción test.

Veremos cada uno de éstos métodos. La figura 1 muestra algunos de los pros y contras de cada método.

Método 1

Si usted tiene acceso a los CDs de instalación para una distribución estándar de Linux, incluyendo Apache como parte de su instalación inicial es indudablemente el modo más fácil a seguir. No tendrá que escribir ningún comando complicado, y terminará con una configuración básica. Así podrá empezar a agregar contenidos Web directamente. Sí, necesitará dedicarle un poco de tiempo descifrando dónde y cómo fue instalado. Por ejemplo, necesitará saber dónde están los directorios **ServerRoot** y **DocumentRoot**, pero eso es sencillo.

Método 2

Si no incluyó Apache en su instalación inicial, su siguiente opción será instalarlo desde el archivo RPM. Necesitará logonearse como `root` para hacerlo. Recuerde que `root` tiene un **#** como prompt.
Recordemos que RPM (Red-Hat Package Management) es un sistema desarrollado por Red-Hat que nos permite saber qué paquetes de software están instalados en la computadora, rastrear las dependencias entre ellos, y manejar muchos de los detalles de instalación y remoción de paquetes. Un "package" es guardado en un archivo `.rpm`. Este incluye los archivos que usa Apache, los scripts para automatizar el proceso de instalación, y la información sobre el paquete y los archivos de los que depende. Por ejemplo, programas ejecutables usualmente dependen (requieren) un número de librerías objeto, con los cuales se conectan dinámicamente cuando empiezan a funcionar. Esta información está en el rpm correspondiente.

Cómo encontrar el archivo rpm en el CD de la distribución

Hacemos:

cd /mnt/cdrom/RedHat/RPMS.

Allí encontrará el archivo llamado **apache-1.3.20-16.i386.rpm**. Si no hubiese sabido por adelantado que debía buscar en el directorio **RedHat/RPMS**, podría haber hecho

find /mnt/cdrom name ´apache*´

ésta es una búsqueda a fuerza bruta, la cual reporta dos archivos:

/mnt/cdrom/RedHat/RPMS/apacheconf-0.8.1-1.noarch.rpm
/mnt/cdrom/RedHat/RPMS/apache-1.3.20-16.i386.rpm

El primero, ´**apacheconf**´, es una herramienta gráfica para configurar Apache, el cual no describiremos aquí.
El segundo es el servidor Apache. Si usted sabe como analizar estos largos nombres de archivo, ellos le dicen mucho acerca del paquete (apache es el nombre del package, 1.3 es la versión, 16 el numero de build, i386 nos dice la arquitectura para la cual es sistema esta compilado: PC en este caso y la extensión .rpm dice que este es un archivo rpm).
Lo primero que debería hacer es usar el comando RPM para hacer una "prueba de instalación" activándolo con la opción test. Esto simula hacer la instalación y reporta cualquier error que pueda encontrar.
En principio todo debería funcionar pero podría darse el caso de que le suceda algo como:

cd /mnt/cdrom/RedHat
rpm -i --test apache-1.3.20-16.i.386.rpm
error: failed dependencies:
libmm.so.11 is needed by apache-1.3.20-16

Lo que esto nos está diciendo es que el paquete Apache necesita un archivo llamado ´**libmm.so.1**´, que no está instalado en la computadora. Entonces dónde busco esta librería **libmm**? Aquí uno podría hacer una suposición incorrecta: que el paquete que contiene el archivo **libmm.so.11** se llamaría **libmm-algo**. Y buscar en los CDs de RedHat, archivos y paquetes RPM empezando por **libmm**. Muy probablemente no encontraría ninguno.
Como opción desesperada podría instalar el Apache con la opción **nodeps**, el cual le dice al RPM que ignore los errores de dependencia:

rpm -i--nodeps apache-1.3.20-16.i368.rpm

El comando se completará correctamente, pero cuando trate de empezar el server, seguramente marcará el error:

httpd -v
httpd: error while loading shared libraries:
libmm.so.11: cannot open shared object file:
No such file o directory

Nota: el modificador **-v** le dice al **httpd** (comando que levanta el servicio web, http) que sólo reporte su número de versión.

Como último recurso, podría ir a www.rpmfind.net/linux/RPM y buscar a **libmm.so.11**. Allí, le informarían que esta en un paquete llamado **mm-1.1.3-8mdk.i586.rpm**. Si ahora regresa a los CDs de RedHat descubrirá que el CD 1 contiene el paquete **mm-1.1.3-1.1386.rpm**.

rpm -qlp mm-1.1.3-1i386.rpm

lo cual confirma que el archivo **/usr/lib/libmm.so.11.0.23** (entre otros) esta incluido. Lo instala, haciendo:

rpm -i mm-1.1.3-1.i386.rpm

Finalmente puede comenzar con el **httpd**:

httpd -v
Server versión: Apache/1.3.30 (Uníx) (Red-Hat/Linux)
Server built: Sp 5 2001 23:12:29

Si quiere aprender como hacer una búsqueda anterior más sintetizada, vea el recuadro "Una alternativa para hallar **libmm**".

¿Puedo hacer lo mismo usando una versión grafica del comando rpm?

Si, hay una versión gráfica del comando RPM llamado GnoRPM. Esto le proveerá de una vista en jerarquías de los RPMs en el CD. Y, le permitirá seleccionar aquellos que usted desee y hacer click en el botón INSTALL. También puede interrogar por los paquetes instalados, por sus descripciones y los archivos que contienen.
GnoRPM no hará nada que usted no pueda hacer con la línea de comando de RPM. De todas formas la línea de comando la necesitaremos para modificar el archivo de configuración y archivos de log (bitácora) de Apache. ▶▷

The Apache Software Foundation
http://www.apache.org/



Apache ya esta instalado:
Ya completa la instalación, veamos los archivos que fueron creados. La opción ql de rpm pregunta por la lista de archivos en el paquete:

rpm -ql apache-1.3.20-16 | less

Note que ahora estamos interrogando los contenidos de un paquete instalado, no un archivo RPM. Algunos de los archivos para destacar incluyen:

* **/usr/sbin/httpd**: el binario del server Apache,
* **/etc/httpd/conf/httpd.conf**: el archivo de configuración de Apache.

El directorio **/etc/httpd** es llamado directorio **ServerRoot**.
Note que la elección de este directorio particular fue hecha (probablemente por quien desarrollo esto en Red Hat) cuando el RPM fue construido. Y, no coincide con el directorio **ServerRoot** asumido durante una instalación Apache por defecto (default) utilizando el código fuente como veremos luego. Los archivos de configuración de Apache, archivos log, y otros archivos necesarios para el funcionamiento, están en este directorio.

También verá un grupo de archivos con nombres como **mod_alias.so**; éstos son módulos compilados lo cuales pueden ser vinculados dinámicamente en Apache para proveer características extras. Finalmente, note el script **/etc/rc.d/init.d/httpd**, este es el startup script que activará el server de Apache (**httpd**) cuando la máquina bootee.

¿Funciona?
Afortunadamente hay suficientes acciones de configuración suministradas por el archivo de configuración que viene por defecto para que Apache comience a funcionar correctamente. Hagamos una rápida mirada a **httpd.conf** (archivo de configuración). Está en **/etc/httpd/conf**, pero recuerde que puede estar en algún lugar diferente, dependiendo de su instalación particular. Es un archivo grande, pero no se asuste. La mayor parte consiste en comentarios de ayuda. Por ahora, hay: solo dos entradas en las que estamos interesados:

ServerRoot "/etc/httpd"
DocumentRoot "/var/www/html"

(Puede ser que usted tenga algunos directorios diferentes nombrados en su archivo). Como vimos **ServerRoot** es en donde vive la configuración y los archivos de soporte **runtime**, y el **DocumentRoot** es el directorio de nivel superior donde uno aloja el contenido Web (los archivos HTML). Por ejemplo, si apunto un buscador Web hacia el URL **http:/localhost/sample.html**, Apache buscará en el directorio **/var/www/html** por el archivo ´**sample.html**´.
Si mira en el directorio **ServerRoot**, aun en una instalación nueva, probablemente encontrará un par de archivos; algo como **index.html** y **poweredby.png**. **Index.html** es un archivo creado por defecto que Apache provee (por ejemplo lo vería si entro en el URL http://localhost) y el archivo png es un gráfico de la pluma del logotipo de Apache. Levantemos el servidor. Lo haremos manualmente, haciendo:

/usr/sbin/httpd

No se asuste si obtiene un prompt de repuesta como si nada hubiese pasado. El servidor automáticamente crea procesos hijos para que corran en el background. Finalmente, usando un browser haga ***http://localhost***. Deberíamos ver la primera página de nuestro servidor.

Iniciando Apache durante el arranque
Supongamos estar satisfechos con la configuración Apache por default recién realizada. Pero antes de que podamos considerar completa nuestra instalación, necesitamos asegurar que el servidor empezará automáticamente cuando el sistema arranque. La instalación RPM puso un script de startup en **/etc/rc.d/init.d/httpd**. Pero necesitamos hacer un link simbólico a este desde el directorio donde esta script de booteo que corresponde al "run level" que usara la máquina. Por ejemplo, si la máquina iniciará en el nivel 5 entonces necesitaré poner un link simbólico en **/etc/rc.d/rc5.d**, algo como :

cd/etc/re.d/rc5.d
ln -s ../init.d/httpd S96httpd

Una alternativa para hallar libmm

Tome el CD 1 de Red Hat, cambie al directorio **/mnt/cdrom/RedHat/RPMS**. Hagamos una búsqueda a fuerza bruta dentro de todos los archivos rpm buscando uno que contenga el archivo llamado **libmm-algo**:

rpm -qlp *.rpm | grep libmm

Lo anterior reportará:

/usr/lib/libmm.so.11.0.23

Esto al menos me dijo que la biblioteca requerida estaba en uno de los archivos rpm del directorio RPMS, pero no me dijo en cuál. Para averiguarlo creemos un pequeño script llamado ´**findrpm**´ como sigue:

```
#!-bin/bash
cd /mnt/cdrom/RedHat/RPMS
for f in *rpm
do
  if rpm- qlp $f l grep libmm
   then
     echo $f
   fi
done
```



Al correrlo obtendrá:

```
# ./findrpm
/usr/lib/libmm.so.11.0.23
mm-1.1.3-1.i386.rpm
```

Esto nos da lo que queríamos: la conexión entre el archivo que estaba buscando y el paquete que lo contenía. Lo anterior hubo que hacerlo ya que no existe una opción para decirle al RPM que busque en todos los paquetes (archivos i.e. rpm) no instalados, el paquete que contiene un archivo dado.
Una de las cosas que enseña lo descrito antes es que no debemos confundir los nombres de los archivos y los paquetes que los contienen: para hacer esto usamos comandos **rpm**, **grep**, y hasta un poco de shell srcipting.

Con este link ahora en su lugar, el servidor de Apache comenzará automáticamente al arrancar la maquina. Aquí asumimos una distribución como Red Hat, la cual usa el estilo de startupscript "System V". Distribuciones como SuSE usan un diseño totalmente diferente.

Método 3: Instalando desde source codes (los códigos fuentes)
Instalar Apache desde los códigos fuente implica compilar más de medio millón de líneas de código, pero no necesitamos ver ninguna de ellas, y la compilación se hace con un solo comando: **make**. Si, necesitaremos tener el compilador C (**gcc**) instalado.
Instalar desde los códigos fuentes tiene varias ventajas. Por un lado, usted puede obtener la última versión del software. Segundo, tener el control de la instalación. Tercero, algunas características importantes (el **secure sockets layer** es un buen ejemplo) sólo pueden ser incluidas a Apache utilizando las fuentes. Una ventaja final es que una vez que usted tiene las fuentes, puede instalar Apache en cualquier plataforma.
Lo que sigue a continuación esta referido a una instalación de la versión beta de Apache 2. (Usted puede bajar la versión que desee desde www.apache.org y la instalación será similar)
Suponga que comienza con el archivo **httpd-2_0_16-beta.tar.gz**. Este archivo contiene lo que generalmente es conocido como ´**tarball**´: un archivo comprimido de los documentos que contiene los códigos fuentes y su estructura de la distribución de Apache. Los tarballs son las formas tradicionales de empaquetar fuentes de distribución para Linux. Antes que sigamos, necesitamos elegir un directorio donde instalar la fuente. Un lugar común para instalar nuevas aplicaciones "locales" es el directorio **/usr/local**. Extraigo el tarball (suponiendo que esta en el directorio **/usr/local**) haciendo:

cd /usr/local
tar xzf /httpd-2_0_16-beta.tar.gz

Este comando descomprime el archivo y extrae los documentos. Si usted busca en **/usr/local** encontrará un nuevo directorio llamado **httpd-2_0_16**, y si mira allí encontrará un documento llamado INSTALL que le dice qué hacer luego. Versiones más viejas de Apache usaban un proceso con su propio archivo de configuración, pero versiones mas nuevas usan exactamente el mismo proceso de configuración que la mayoría de otras distribuciones fuentes. Simplemente vaya al directorio del que extrajo los documentos y ejecute el comando:

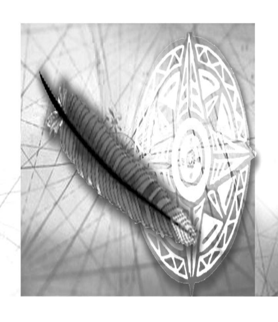
./configure

Esto es un shell script. Fue creado por un

Afortunadamente hay suficientes acciones de configuración suministradas por el archivo de configuración que viene por defecto para que Apache comience a funcionar correctamente.

programa llamado autoconf (una de las herramientas de GNU) cuando el tarball fue creado. El script de configuración prueba que el sistema verifique la presencia de varias características de las librerías y compiladores del sistema. Basado en este descubrimiento, creo algunos makefiles y shell scripts que construirán el software en su plataforma particular. Esta pieza de tecnología hace mucho mas fácil el crear una fuente de distribución que se instale sobre un espectro amplio de sistemas *nix. (Linux, Unix Aix de IBM, Unix Solares de Sun, etc.)

El siguiente paso es compilar las fuentes de Apache. El script de configuración ha creado los makefiles y scripts necesarios para controlar este proceso. Todo lo que se debe hacer es tipear la palabra "**make**". Debemos esperar bastante ya que hay muchísimo código fuente por compilar. En una máquina Pentium II puede tomar alrededor de 10 minutos.



Finalmente para instalar Apache (es decir distribuir los ejecutables en sus directorios etc.) hago:

make install

Esto no toma mucho tiempo

Nota de Interés: ¿Cuánto código fuente hay? Bueno, como otra ilustración del poder de las herramientas de Linux usadas en forma combinada, este comando se lo dirá:

cd /usr/local/httpd-2_0_16
wc -l `find . -name `*.c´-print`

La respuesta, es un poquito más de 500,000.

Para resumir, existen cinco pasos para instalar un tarball:

1-desempacar el tarball dentro de un directorio elegido
2-hacer cd a este directorio
3- ***./configure***
4-make
5-make install

Una ventaja final es que una vez que usted tiene las fuentes, puede instalar Apache en cualquier plataforma.

Figura 1: Pros y contras de los métodos de instalación

Método de instalación	Ventajas	Desventajas
Como parte de la instalación de la distribución de Linux	Fácil, probablemente funcionará	Usted tiene que decidir si necesitará levantar el servidor. Usted obtendrá la versión que esté en la distribución, la cual probablemente será vieja. Usted no tiene los códigos fuente.
Instalado desde un RPM binario	Rápido y fácil, no hay problema con perder las dependencias. Usted puede bajar e instalar una versión más nueva que la de la distribución original. El RPM maneja la administración de dependencia por usted.	Usted no obtendrá la última versión. Usted tiene flexibilidad limitada en configurar la instalación. Usted no tiene las fuentes. Necesita un RPM diferente para cada arquitectura.
Instalado desde una tarball fuente	Usted puede obtener la última versión. Tiene total flexibilidad en configurar la instalación. Tiene el código fuente.	Toma un poco más de tiempo. Necesita ayuda experta si la construcción falla. Un tarball maneja todas las arquitecturas.

Existen varias formas de customizar el proceso de instalación. Por ejemplo, podríamos suministrar una opción como:

./configure prefix="/usr/local/apache2_0_16"

en orden de ubicar expresamente el directorio **ServerRoot**. Si aceptamos el default sería ***/usr/local/apache2***.

Note que instalar de esta forma desde la fuente no actualiza la base de datos de RPM de los paquetes instalados. Por ejemplo, si yo usare **rpm -qa** para interrogar todos los paquetes instalados, no vería mi **apache-2.0.16** recientemente instalado. Tampoco podría usar el RPM para removerlo.

Testeando el web Server

Pruebe su server iniciándolo manualmente, como hicimos después de la instalación de RPM (excepto que ahora el binario es **/usr/local/apache2/bin/httpd**). Luego usted podrá poner una entrada dentro de un script boot-time apropiado.

Por ejemplo agregamos lo que sigue al startup script en **/etc/rc.local**

echo starting apache 2
/usr/local/apache2/bin/httpd

Esto debería funcionar bien, aunque no veríamos ninguno de los mensajes del sistema, sobre empezar y detener servicios como sucede al usar el rpm de Red Hat. Advertencia: Ni siquiera intente empezar más de una versión de Apache. Porque, si lo hace, ellos terminarán compitiendo por el puerto 80 y el que empiece segundo, indudablemente terminará perdiendo.



Preguntas LPI

Examen LPI 101

¿Cuál de los siguientes comandos sería el mejor para montar un CD-ROM conteniendo una distribución Linux, sin depender de ningún archivo de configuración?

a) *mount /dev/cdrom/dev/hdc*
b) *mount -f linux/dev/hdc/mnt/cdrom*
c) *mount -t iso9660 /dev/cdrom /mnt/cdrom*
d) *mount -t linux /dev/cdrom /mnt/cdrom*
e) *mount -t iso9660 /mnt/cdrom /dev/cdrom*

Rta: **C**. Los CD-ROM usan los sistemas de archivos iso9660, el cual es el defecto del montaje, pero también indica el uso de *-t*. Sin la asistencia de una entrada en */etc/fstab*, los dos dispositivos de montaje (*/dev/cdrom*) y el punto de montaje (*/mnt/cdrom*) deben ser proveídos.

Examen LPI 102

Carolina tiene un archivo de texto llamado *guest_list* que contiene 12 líneas. Ella ejecuta el siguiente comando:

$ split -4 guest_list gl

¿Cuál es el resultado? Seleccione uno

a) Las primeras cuatro columnas en el texto son escritas a archivos nuevos *glaa*, *glab*, *glac* and *glad*
b) Las primeras cuatro columnas en el texto son escritas a archivos nuevos *aagl*, *abgl*, *acgl* and *adgl*
c) Las líneas *guest_list* están igualmente divididas entre archivos nuevos *glaa*, *glab*, *glac* and *glad*
d) Las líneas de *guest_list* están igualmente divididas entre archivos nuevos *glaa*, *glab* and *glac*
e) Las líneas de *guest_list* están igualmente divididas entre archivos nuevos *aagl*, *abgl*, and *acgl*

Rta: **D**. "*split -n file outfile*" separa un archivo en archivos de salidas múltiples, cada uno con *n* líneas y los nombran *outfileaa*, *outfileab*, y así sucesivamente. Como el archivo original tiene 12 líneas y Carolina las dividió en sets de 4 líneas, el resultado es 3 archivos llamados *glaa*, *glab* y *glac*, cada uno conteniendo 4 líneas.

Mozilla: Como adaptarlo a nuestras necesidades

En la actualidad existen varios browsers para navegar en Internet (ya sea corriendo bajo Windows o Linux): Netscape, Internet Explorer (sólo corre bajo Windows®, y Mac.), Opera, Mozilla. Mozilla aparte de muy popular es excelente. Está dando al escritorio Linux un profesionalismo como lo han venido haciendo OpenOffice.org, KDE o GIMP.



Lo interesante de Mozilla es que además de tener una buena funcionalidad, es muy maleable (adaptable a nuestros requerimientos). Esto significa, por ejemplo, que escuelas, ISPs (proveedores de Internet) y oficinas pueden adaptar una aplicación entera de Internet a las necesidades de sus estudiantes, clientes o empleados, o proveer un producto de mercado (con un nivel de complejidad mayor o menor dependiendo de la funcionalidad requerida) sin necesidad de contratar a un equipo costoso de artistas y de desarrolladores.
Lo que intentaremos hacer en este artículo es mostrarles como crear un browser hecho a medida. Esto lo haremos ampliando las funcionalidades del browser sin necesidad de modificar codificación XML y comenzaremos por una aplicación de desarrollo bien básica. Hackear Mozilla (hackear se entiende como modificar, cambiar) puede ser tan simple o complejo dependiendo del interés que tengamos de meternos en los archivos de configuración.

Comenzaremos con algunos ajustes de menor importancia. Pero también es posible realizar cambios fundamentales con apenas algunos cambios en algunas líneas de código. Tampoco necesitaremos tener un ambiente de desarrollo completamente integrado para realizar y evaluar los cambios (aunque si lo deseamos podemos utilizarlo). Un editor de textos básicos, un selector de color y un paquete capaz de manejar gráficos deberían ser suficientes. Siéntase libre de utilizar el programa que le sea más cómodo. Si puede salvar el archivo de configuración como .css es suficiente (css viene de Cascading Style Sheets). Lo que haremos se llama cambiar la piel del browser (en ingles "skinning").

Punto de partida.

Partiremos de un skin base ya desarrollado. Comenzar un skin desde cero puede resultar bastante largo y tedioso y quizás sin sentido. Comenzaremos utilizando el tema moderno (modern) (ya que se parece al browser que deseamos diseñar), por supuesto que podemos usar el Clásico (classic), o descargar algún otro de la página www.mozdev.org para basar nuestras creaciones. O si tiene el tiempo e inclinación, puede comenzar sin ninguno. Si tiene miedo que Mozilla no vuelva a arrancar luego de sus cambios sugerimos tener una segunda instalación en su Home directory. Esta es una buena metodología para hacer pruebas y además no tenemos que lidiar con temas de permisos.

Averiguemos qué se ejecuta cuando abrimos la aplicación Mozilla. Abra una terminal y escriba:

which mozilla

le responderá algo como

/usr/bin/mozilla

es decir, mozilla (en el directorio */usr/bin/*) es el binario (ejecutable) buscado.
Ahora, averiguemos cual es el directorio que contiene todo lo relacionado a mozilla (programas, archivos de configuración, script files etc).

*# find / -name mozilla**

Entre la respuesta vera algo como:

/usr/lib/mozilla-1.2.1

Entonces para poder trabajar tranquilos modificando etc, copiemos ese directorio a nuestro directorio home.

cp /usr/lib/mozilla-1.2.1 $HOME/

Vayamos al directorio para investigarlo:

cd $HOME/mozilla-1.2.1

ls

nos mostrará los directorios y archivos. Alli vera por ejemplo: *run-mozilla.sh*. Ese es el ejecutable (idéntico al que esta /usr/bin/) que lanza mozilla. Pruébelo!!!, haga:

./run-mozilla.sh

Ahora, pongamos a Mozilla a usar el skin modern. En la ventana que recién abrió mozilla, vaya a view/themes, indique modern (la alternativa es classic). Cierre Mozilla y córralo otra vez. Vera como cambio su look.

Comenzar un skin desde cero puede resultar bastante largo y tedioso

Lo primero que debemos entender es que Mozilla ha sido configurado desde sus primeras versiones para usar skins guardados en archivos .jar. Estos son básicamente archivos comprimidos disimulados con una extensión diferente a las populares; de modo que podemos abrirlo con cualquier herramienta de compresión, aunque, quizás tenga que renombrarlo como "modern.zip" si su utilitario de compresión no reconoce los archivos .jar.
Por ejemplo si va al directorio */mozilla-1.2.1/chrome/* alli vera el archivo modern.jar. Extráigalo a */chrome/modern/* pero mantenga el resto de la estructura de directorio:

mkdir $HOME/mozilla-1.2.1/chrome/modern

cd $HOME/mozilla-1.2.1/chrome/modern

#cp ../modern.jar .

#unzip modern.jar

Tómese un momento para mirar alguno de los contenidos del directorio *mozilla-1.2.1*. Encontrará una colección de archivos GIFs, CSS, DTD, varios documentos XML, y algunos otros archivos organizados lógicamente: /global es usado para configurar botones, dialog-boxes, barras de desplazamiento y similares; */comunicator* es el depósito para el barra de estado, barra de tareas, y perfil de información y las otras son aplicaciones específicas correspondientes al browser, messenger y composer.

Ahora bien, podríamos inmediatamente entrar y empezar a cambiar cosas, pero a pesar de los cambios no se produciría ningún cambio en el browser. Esto es porque Mozilla todavía esta mirando al *modern.jar* para la información de su configuración. Para modificarlo necesitamos editar el archivo */mozilla-1.2.1/chrome/installed-chrome.txt* con un editor de textos y buscar la siguiente sección que necesitaremos modificar:

Skin,install,url,jar:resource:chrome/modern.jar!/skin/modern/communicator/
Skin,install,url,jar:resource:chrome/modern.jar!/skin/modern/editor/
Skin,install,url,jar:resource:chrome/modern.jar!/skin/modern/global/
Skin,install,url,jar:resoruce:chrome/modern.jar!/skin/modern/messenger/
Skin,install,url,jar.resource:crome/modern.jar!/skin/modern/navigator/

Simplemente sacamos las referencias jar ligadas a modern quedando de la siguiente manera:

Skin,install,url,resource:/chrome/modern/skin/modern/communicator/
Skin,install,url,resource:/chrome/modern/skin/modern/editor/
Skin,install,url,resource:/chrome/modern/skin/modern/global/
Skin,install,url,resource:/chrome/modern/skin/modern/messenger/
Skin,install,url,resource:/chrome/modern/skin/modern/navigator/

Puede llegar a suceder que hayamos hecho los cambios y no notar ninguna diferencia, en ese caso, tendremos que deslizarnos hacia abajo en el documento y encontrará otra sección que todavía estará señalando a los archivos .jar. Cámbielo como señalamos anteriormente.
Para demostrar alguno de los fundamentos de skining, lo que haremos será alterar el menú bar: modificamos el fondo a naranja y cuando se desplieguen las opciones también serán naranjas. Cuando el mouse se deslice remarcara con rojo.
Los resultados no son enormemente espectaculares pero muestran como unos pocos cambios pueden hacer a una gran diferencia.

Skinning

Para comenzar a preparar nuestro skin, usaremos estos

▶

Formato de las imágenes

La base fundamental de Mozilla.org es que todo lo que se dibuja dentro de la aplicación esta hecho por Gecko, es decir que al diseñar skins no estamos restringidos a usar imágenes GIF, a pesar de que los skins Standard están repletos de ellas. Los archivos GIF son útiles (pequeños y con capacidad de transparencia) pero es una lastima perder todo el poder de diseño. Recordemos que las imágenes están almacenadas localmente y que no necesitan ser cargadas desde conexión de red.
Una mejor alternativa es usar archivos PNG para los gráficos.
Mientras que el tamaño de los archivo es reducido, los PNGs no se ven tan borrosos como los GIFs, por lo cual sus botones y fondos pueden tener una mejor definición y también puede usar transparencias tipo-alpha, que añadirán sombras realistas y luminosidad a su trabajo. El futuro traerá aún mejores opciones ya que Mozilla tendrá un subsistema de rendering nativo SVG. Esto significa que muy factiblemente llegue a tener una UI (interface de usuario) de plataforma cruzada, totalmente escalable muy parecido a lo que hoy hace el tema Gorrilla de Ximian.

tonos de naranja como la base de nuestro tema. Primero editemos /global/toolbar.css y vayamos a la sección marcada como "menubar" donde podremos configurar el background (fondo) y los bordes del menubar.

```
Menubar {
  cursor: default;
  min-width: 1px; /*"DONT DELETE!
  Prevents hiding of scrollbars in browser
when window is made smaller*/
  background-color: #FFCC99;
  color: #000000;
}
```

Hemos modificado el background-color (color de fondo) y color, configurando el fondo y el color del texto del menú.
Así, hemos cambiado el tradicional azul grisáceo por un anaranjado. (usando los colores del Websafe sacados de Kcolour) mientras que el texto se mantiene en negro. Recuerde que, si hacemos cambios en directorio /global, cualquier cosa que cambie se verá reflejado en las otras aplicaciones del browser. También modifiquemos:

```
menubar>.toolbar-holder{
  Border-top: 1px solid # FF9933;
  Border-right: 1px solid # FFCC66;
  Border-bottom: 1px solid # FF9900;
  Border-left: 1px solid # FF6600;
}
```

En los cambios anteriores se definen las áreas que rodean el menú bar; en este caso los cuatro bordes tienen un solo píxel de ancho y tonos mandarina. Si usted lanza Mozilla ahora, el menú bar y sus bordes estarán coloreados, pero al clickear alguna opción de la barra de menú se desplegará el clásico gris. Veamos como modificar esto último:
Abra ahora */global/menu.css* y encuentre la sección marcada ***/*:::::menu/menuitem::::: */***. La primer referencia de color que usted verá aquí, define el texto del menú desplegadle, mientras que el segundo pone el color a los ítems del menú que no están disponibles. Como no hay color de fondo en esta primera sección menuitem, añada la línea: ***background-color:#FFCC99;***

Para cambiar el color de los ítems del menú cuando uno desliza el mouse sobre ellos encuentre ***/* ::::: menú/menuitems in popups ::::: */*** Aquí vera las dos (tags) etiquetas familiares: ***background-color*** (color de fondo) y ***color***, que cambian los colores de posicionamiento y los del texto respectivamente. Los cambios podemos chequearlos en un instante, así que adelante! Guarde su archivo, cierre, vuelva a abrir Mozilla, debería tener algo similar a la Figura 1, con los colores naranjas y rojos.

Aclaremos, que todo lo que hemos configurado usando colores de referencia hex (i.e ***#FFCC99***). Pero, pueden también ser hecho con una imagen de fondo, Justamente por eso en la figura 1 tenemos una barra de herramientas grisácea con un borde color anaranjado no bien definido. O que sucede es que al usar el modelo CSS, el estilo del browser (la barra de herramientas con una imagen) sobrescribe el estilo global (la barra de herramientas con referencia hex), pero una diferencia en el tamaño de la imagen provoca un borde desagradable en el fondo de la sección.
Si desea entender es esto y realizar las modificaciones abra ahora */communicator/toolbar.css* y después descienda en el documento pasando la licencia a la sección toolbar-primary. El tinte graduado en la barra de herramientas es llamado por la siguiente línea:
background-image: url("chrome://communicator/skin/toolbar/prtb-bg-noline.gif") repeat-x top;
Aquí tenemos dos opciones, o bien apuntar el URL a un archivo diferente en tal caso, crearemos en el GIMP un archivo grafico nuevo, de un píxel de ancho x 40 pixels de profundidad, lo coloreamos a gusto y cambiamos las referencia en el archivo CSS, o simplemente podemos cambiar la imagen existente. De cualquier manera, el resultado es una imagen repetida corriendo la longitud (repeat-x) de la barra de herramientas.

La idea del ejercicio anterior era enseñarnos cómo es de fácil modificar skins. La página web *www.mozilla.org* esta llena de referencias, HOW-TOs y links que nos permitirán ser expertos en este arte si así lo deseamos. Es importante saber que existe una solución más simple para nuestra "Adaptación de Mozilla", lea el apartado sobre el proyecto Phoenix.



Figura 1

Si hacemos cambios en directorio/global, cualquier cosa que cambie se verá reflejado en las otras aplicaciones del browser

Los archivos ".jar" son básicamente archivos comprimidos, disimulados con una extensión diferente a las populares.



Proyecto Phoenix. Un Mozilla de fácil customización.

Tratar de cambiarle totalmente el aspecto a Mozilla puede ser complejo. Esto es porque hay muchos archivos de imágenes y referencias que necesitan construirse y editarse. Si solo se necesita darle una apariencia personalizada al browser (para navegar en una empresa/universidad o acceder al webmail), entonces aquí podemos tener una ayuda considerable de parte de "la hermana menor de Mozilla", Phoenix.
Phoenix es un explorador de internet similar a Galeon, pero con la ventaja de que esta integramente construido en XUL, Mozilla´s eXtensible User interface Language. De modo similar a lo que describimos en este articulo, modificar una distribución personalizada de Phoenix, es muy fácil para cualquier persona que sea capaz de cambiar algunas líneas de códigos y diseñar u agregar algunas imágenes. Para instalarlo simplemente haga un download desde *http://www.mozilla.org/projects/phoenix*, y descomprímalo dentro de un directorio (así es, la instalación es tan simple como eso). Necesitará descomprimir nuevamente su skin base (Phoenix solamente viene con el skin Classic) dentro del directorio skins y después editar installed-chrome.txt para apuntar al skin descomprimido. A pesar de descomprimir una gran cantidad de gráficos y archivos CSS en su disco rígido, la mayoría del trabajo hecho en Phoenix toma lugar en el directorio /browser. Aquí encontrará una selección de archivos de gráficos y un único archivo CSS (que confusamente también es llamado browser). Phoenix hace fácil el empaquetar y distribuir sus skins, lo que ha provocado un número creciente de skins en Websites de terceros tales como *http://mozdev.org* (una comunidad de desarrolladores on line que saturó el número de skins oficiales disponibles del propio Mozilla a solo unas semanas del lanzamiento de la versión 0.3)

La Tecnología Atrás de Mozilla

Un pantallazo a la tecnología que hace que Mozilla sea tan flexible

En el contexto de desarrollo de sitios Web, el Cascading Style Sheets, CSS permiten al Webmaster definir varios elementos de su sitio en un solo documento. Por ejemplo, fondos (backgrounds), colores y formatos de fuentes. Simplemente poniendo un link en el ***<head>*** de una pagina HTML de ese documento forzaría a la pagina a heredar los estilos definidos previamente. La mayor ventaja es que usted puede refinar su diseño y hacer cambios en la totalidad del site alterando solo un documento. Separa así el diseño del contenido
Si quisiera que una pagina específica tuviese una apariencia distinta, puede agregar información del formato y estilo al **<head>** de los documentos individuales y esto eliminaría la referencia a la hoja de estilo externa.. Además, si quisiera modificar el estilo de un solo elemento de la pagina podría modificarlo de la forma normal y este eliminaría el formato de la referencia externa y la del ***<head>*** de la pagina. Por eso el término "cascading" (cascada). Lo fundamental a recordar es que el formato más cercano al elemento que será modificado toma precedencia.
Así también es como trabaja la UI (User Interface) de Mozilla Si en el directorio */global* editamos *toolbar.css* para seleccionar un color sólido para el fondo (background) de la barra de herramientas, después definimos en el directorio */communicator* la imagen del fondo para la barra en *toolbar.css*, y después en la carpeta */navigator* modificamos las imágenes de los botónes, será este ultimo cambio el que prevalecerá. Mientras que el directorio */communicator* sería el "más cercano" al elemento UI que hemos diseñado, prevalecerá lo que modifiquemos en /navigator ya que es el que se encuentra mas cerca aún. Se han escrito libros extensos acerca de CSS, y explicarlo completamente esta mas allá del alcance de este articulo, pero la comprensión de la tecnología lo ayudará mucho a trabajar con Mozilla.
Un primer acercamiento puede ser encontrado en: *www.webdesignhelper.co.uk/design_elements/css/css3/css3.shtml*. Allí, encontrará la definición oficial de CSS en *www.w3.org/TR/REC-CSS1*. Mientras que CSS definirá la apariencia de su proyecto de browser, el funcionamiento de los botones, la barra de herramientas y los menús están basados en otra tecnología de Web Standard llamada Javascript.



ASP vs PHP

Los lenguajes ASP y PHP fueron creados con el fin de permitir a los diseñadores y programadores de páginas web interactuar de una forma más eficiente con los usuarios. Ambos lenguajes tienen cosas en común, entre ellas, un gran número de adeptos y desertores. Tanto uno como otro permiten realizar todo tipo de aplicaciones web, desde sencillos scripts como ciclos de decisión o repetición, hasta cosas más complejas, como acceso a bases de datos, manipulación de archivos de texto y binarios, y mucho más.

La ventaja de ASP es que es bastante más descriptivo en caso de encontrar algún error en el código

Una enorme ventaja de PHP es que está basado casi totalmente en funciones

Los Lenguajes

Tanto PHP como ASP son lenguajes de script. Esto significa que una aplicación desarrollada con cualquiera de estos lenguajes no es más que un simple archivo de texto, en donde se encuentra el código fuente original, escrito por el programador. A diferencia de otros lenguajes, no se compilan, sino que se "interpretan" por el servidor web.

Generalmente, se asocia a PHP con Apache y el mundo UNIX, y a ASP con Internet Information Server (IIS) y el mundo Windows. Sin embargo, las cosas no son ni blancas ni negras. PHP tiene versiones para una gran cantidad de sistemas operativos basados en UNIX, y también para Windows; a su vez, puede ser instalado tanto sobre Apache como sobre IIS. Apache también tiene versiones muy estables para sistemas de tipo UNIX y para Windows. Por el lado de ASP, ya nos quedamos más cortos; Microsoft sólo desarrolla ASP y su servidor IIS para Windows NT, XP o 2003. Si bien existe una versión de ASP para Linux (Chilli ASP), su utilización puede resultar más problemática que pasarse a un servidor NT con todas las ventajas que ello implica.

Funcionamiento

En el gráfico HTML vs. PHP/ASP podemos ver cómo funcionan estos lenguajes. Se puede observar que cuando un cliente (el navegador de un usuario) hace un pedido de una página ASP o PHP al servidor, primero éste la procesa, en lugar de enviarla directamente al navegador. La diferencia con respecto a una página común es que un archivo HTML no pasa por el intérprete de ASP o PHP antes de ser enviado al navegador. Básicamente, éste es el modo en el que trabaja la mayoría de los lenguajes de script ideados para Internet.

¿Qué me conviene usar?

Seguramente ésta sea la pregunta que se harán tanto quienes están decididos a aprender programación web como quienes están evaluando el desarrollo de un nuevo sitio. La respuesta depende de un montón de factores, que habrá que analizar con cuidado.

En primer lugar, si necesitamos alguna tecnología específica, deberemos evaluar qué lenguaje nos la brinda. Las posibilidades aquí son infinitas, aunque se trata de un caso muy correcto y particular.

Si la decisión pasa por lo económico, vayan sabiendo que instalar y mantener un servidor web con Linux, Apache, PHP y MySQL es muchísimo más barato que hacerlo con Windows 2000 o 2003, Internet Information Server y SQL Server 2000. Las diferencias están en el orden de miles de dólares. Del mismo modo, y por la misma razón, los servicios de hosting bajo Linux son notablemente más económicos que sobre Windows, por lo que el punto más importante ya está a favor de PHP.

En cuanto a la estabilidad de una aplicación diseñada con ASP o PHP, podemos asegurar que no tendremos demasiados problemas. La ventaja de ASP es que es bastante más descriptivo en caso de encontrar algún error en el código. PHP, en cambio, puede tirarnos errores con referencias a líneas que están correctamente programadas, pero que son afectadas por un error arrastrado de una línea anterior.

Una enorme ventaja de PHP es que está basado casi totalmente en funciones. El lenguaje provee una enorme librería de funciones para las tareas más diversas: desde funciones para trabajar con el sistema de archivos, con vectores, con cadenas de texto y con fechas y horas, hasta algunas menos comunes, pero muy útiles para manipular imágenes, comprimir archivos, crear PDFs, acceder a bases de datos MySQL, SQL Server 2000, Oracle, Postres SQL, Sockets, FTP, y muchísimas más.

ASP, en cambio, es un lenguaje más orientado a objetos, aunque no dispone de tantas funciones predefinidas como PHP. Para gran parte de las tareas menos frecuentes, es necesario instalar componentes adicionales (por ejemplo, para subir archivos al servidor o para manipular imágenes). Esto es verdaderamente una molestia, dado que si estamos empleando un hosting comercial, no será muy común que nos permita instalar algo.

Uno se puede sentir más cómodo trabajando con PHP. Si bien su sintaxis no es tan literal como ASP, se suelen requerir menos pasos para realizar las mismas tareas que en el lenguaje de Microsoft, lo que hace que el código sea más sencillo y breve. Quizás uno tenga que programar sitios utilizando ASP, y se vea en la necesidad de emplear componentes de terceros para subir archivos ▶

Tareas Especiales



Aquí veremos como se realizan las mismas tareas en ASP y en PHP respectivamente.

ASP

```
<%
De = "correo@dominio"
Para = "correo@dominio"
Asunto = "Asunto"
Texto = "Texto del e-mail"
Set mail =
Server.CreateObject("Persits.MailSender")
mail.Host = "ServidorSMTP"
mail.From = De
mail.AddAddress Para
mail.Subject = Asunto
mail.IsHTML = True
mail.Body = Texto
mail.Send
%>
```

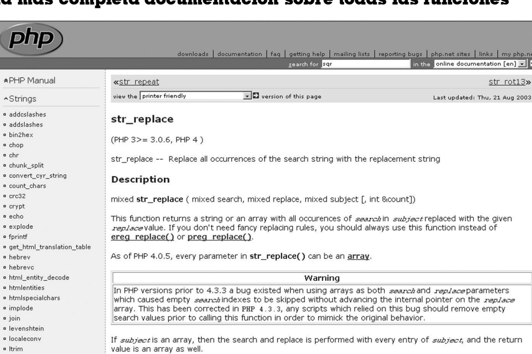
PHP

```
<?
$Para = "correo@dominio";
$Asunto = "Asunto";
$Texto = "Texto del e-mail";
$Encabezados = "From: Nombre
<correo@dominio>\r\n";
$Encabezados .= "Content-Type: text/html;
charset=iso-8859-1\r\n";

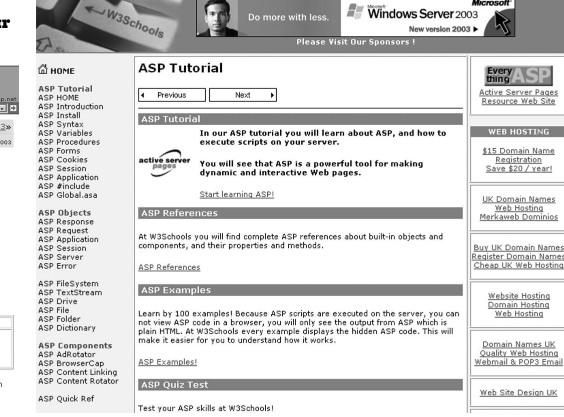
mail ($Para, $Asunto, $Texto,
$Encabezados);
?>
```



En el sitio oficial de PHP (www.php.net) podremos encontrar la más completa documentación sobre todas las funciones



Microsoft no ofrece un sitio similar para ASP, ni tan bien organizado como el de PHP, pero podremos encontrar ayuda en www.w3schools.com/asp.



o enviar e-mails. Si, por ejemplo, debemos cambiar el sitio de proveedor de hosting, y se encuentra con que el nuevo ISP utiliza componentes diferentes del anterior, seguramente deberá reprogramar todo. Si bien es sólo un detalle, son características importantes en un lenguaje para el fin que persigue y que no pueden dejarse sin cuidado.

¿Por dónde comenzar?
Si hemos elegido ASP, ya sea para aprender o para trabajar, y tenemos instalado Windows XP o Windows 2000, no tendremos demasiados problemas de configuración e instalación. Para poder empezar a trabajar, es necesaria la instalación de Internet Information Server (IIS), el servidor web de Microsoft. Para ello, si es que no lo hicimos antes, debemos dirigirnos a [Agregar o quitar programas] y seleccionar [Agregar o quitar programas]. En la ventana que aparece, debemos marcar la opción [Servicios de Internet Information Server (IIS)] y seguir los pasos del asistente. Una vez que hayamos instalado este conjunto de servicios, ya podremos comenzar a utilizar ASP.
Si, en cambio, nuestra elección se inclinó hacia el lado del software libre, entonces tenemos dos opciones: instalar PHP sobre IIS, o bien instalar primero Apache como servidor web y, luego, PHP. De estas dos opciones, la segunda es la más recomendable, para aprovechar todas las características de PHP. Cabe aclarar que Apache puede instalarse tanto sobre Linux como sobre cualquier versión de Windows, a diferencia de IIS, que sólo puede instalarse sobre Windows 2000, XP o 2003.

Fuente: PC Users

Apache puede instalarse tanto sobre Linux como sobre cualquier versión de Windows, a diferencia de IIS, que sólo puede instalarse sobre Windows 2000, XP o 2003.

Diferencias fundamentales entre los lenguajes

	ASP	PHP
Palabras clave	Es indistinto; pueden ir en mayúsculas o en minúsculas.	Todas las funciones y las palabras clave se escriben en minúscula.
Sensible a may. o minúsculas	No	Si
Fin de instrucción	Cambio de línea	Punto y coma (;)
Comentar una línea	'Comentario	//Comentario
Comentar un bloque	'Hay que comentar 'cada línea por separado	/*Comentario */
Variables	No es necesario declararlas	No es necesario declararlas Se identifican con un signo $ delante
El código se escribe entre las etiquetas:	<% y %>	<? y ?>
Permite conectarse a orígenes de datos ODBC	Si	Si
Conexión a SQL Server sin ODBC	Si	Si
Conexión a MySQL sin ODBC	No	Si
Puede correr sobre Linux	No	Si
Puede correr sobre Windows	Si	Si

Diferencias de código: Tabla Comparativa

Ambos lenguajes permiten realizar tareas similares; es importante conocer y saber diferenciar bien uno de otro para no confundirse. En la siguiente tabla comparativa se explica cómo realizar algunas de las tareas más frecuentes en uno y otro lenguaje:

Tarea	ASP	PHP
Enviar texto al navegador	<% Response.Write("Texto") %>	<? echo "Texto"; ?>
Forma abreviada	<%="Texto"%>	<?="Texto"?>
Concatenar cadenas	<%Response.Write(cadena1 & cadena2)%>	<? echo $cadena1 . $cadena2; ?>
Escribir una cookie que expire en 24 días	<% Response.Cookies("nombre")=Valor Response.Cookies("nombre").Expires= dateadd("d",24,Date) %>	<? Setcookie ("nombre", valor, time()+86400*24); ?>
Leer una cookie	<%valor=Request.Cookies("nombre")%>	<?$valor=$_COOKIES["nombre"];?>
Leer una variable de URL (GET)	<%valor=Request.QueryString("nombre")%>	<?$valor=$_GET["nombre"];?>
Leer una variable de un formulario (POST)	<%valor=Request.Form("nombre")%>	<?$valor=$_POST["nombre"];?>
Redirigir a otra página	<%Response.Redirect("pagina.asp"%>	<?header("Location: pagina.php");?>
Finalizar la ejecución de un script	<%Response.End()%>	<?exit();?>
Los componentes de una fecha	<% Dia = Day(Date) Mes = Month(Date) Ano = Year(Date) %>	<? $Fecha = getdate(); $Dia = $Fecha["mday"]; $Mes = $Fecha["mon"]; $Ano = $Fecha["year"]; ?>
Los componentes de la hora	<% Hora = Hour(Time) Minutos = Minute(Time) Segundos = Second(Time) %>	<? $Fecha = getdate(); $Hora = $Fecha["hours"]; $Minutos = $Fecha["minutes"]; $Segundos = $Fecha["seconds"]; ?>
Reemplazar texto en una cadena	<% NuevaCadena = Replace(cadena, "txtBuscar", "txtReemplazar") %>	<? str_replace("txtBuscar", "txtReemplazar", cadena); ?>
Separar una cadena de elementos de un vector, de acuerdo con un delimitador	<% vector = Split(cadena,delimitador) %>	<? $vector=explode(delimitador,cadena); <?

Microsoft Developer Network (msdn.microsoft.com) puede ser otro sitio para obtener información y ayuda sobre desarrollo con ASP.



HTML: Estático vs Dinámico

En estos gráficos observamos la diferencia en el funcionamiento de los servidores, respecto de los lenguajes de programación de páginas, ya sea HTML para Páginas Estáticas ó ASP/PHP para Páginas Dinámicas.

Petición de Páginas Estáticas

Navegador → pág.html → Servidor → Pág. con código HTML → pág.html → Navegador

Petición de Páginas Dinámicas

Navegador → pág.php → Servidor Apache → Pág. con código PHP → Módulo PHP → Pág. con código HTML → pág.php → Navegador

Preguntas Examen 70-228: Administración de SQL Server 2000

Usted esta ejecutando actualmente SQL Server 7. Sabiendo que SQL Server 2000 es capaz de correr múltiples instancias del mismo servidor: ¿Cuál de las siguientes afirmaciones es válida?
a-Si SQL Server 2000 coexiste con otra versión, la instancia de SQL Server 2000 debe ser la instancia por defecto.
b-SQL Server 2000 sólo puede coexistir con otras versiones de SQL Server.
c-SQL Server 2000 puede ser ejecutado en un SQL Server 7 ya existente, dejando intacto el Server existente.
d- SQL Server 2000 sólo puede coexistir con otras instancias de SQL Server 2000.

Rta: C. No existe una limitación respecto de las versiones de SQL Server con las que SQL Server 2000 puede coexistir. La única limitación es que las versiones no-2000 solamente pueden ser 6.5 ó 7.



PERIÓDICO DE NETWORKING
Y PROGRAMACIÓN

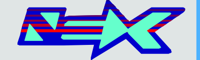
NEX Mensual

A PARTIR DE FEBRERO 2004

www.nexweb.com.ar

Distribución Gratuita

CON UNA TIRADA INICIAL MENSUAL DE 20.000 EJEMPLARES



Han vuelto los reportes en los exámenes Microsoft

Microsoft ha re-introducido los "score reports" (reporte sobre resultados) de los exámenes dentro del programa de MCP. Según un FAQ publicado en su página web, Microsoft re-implementará la entrega de un reporte sobre los resultados (score reports) a los examinados en el final de los exámenes. Los score reports serán implementados para los nuevos exámenes relacionados con Windows 2003. Pero, tiene esperanzas de proporcionarlos para todos los exámenes para el final de septiembre.

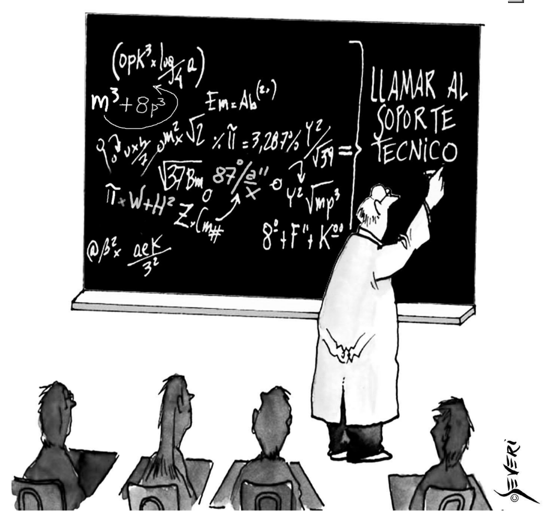
De acuerdo con las FAQ, el resultado del examen incluirá "una cuenta numérica y un gráfico de barras para cada sección del examen". Las habilidades (skills) que aparecen en el informe se correlacionarán con las que Microsoft publica en las guías de objetivos para el examen.

En las FAQ además se menciona que Microsoft ha desarrollado una escala para el examen con un resultado mínimo de 700; el máximo continuará variando dependiendo de varios factores, incluyendo la duración del examen, la complejidad de la pregunta y otras medidas psicometrías.

Hace tiempo Microsoft había implementado un reporte muy simple de los resultados. Estos daban una evaluación numérica por sección. La compañía decidió omitir resultados numéricos en Junio de 2000, cuando estaba implementando sus exámenes de Windows 2000. Dice que está re-implementando el brindar los resultados de exámenes basándose en el feedback con los clientes.

Este reporte sobre los resultados de los exámenes es apenas uno de varios cambios que se pondrán en ejecución en el programa de certificaciones de Microsoft.

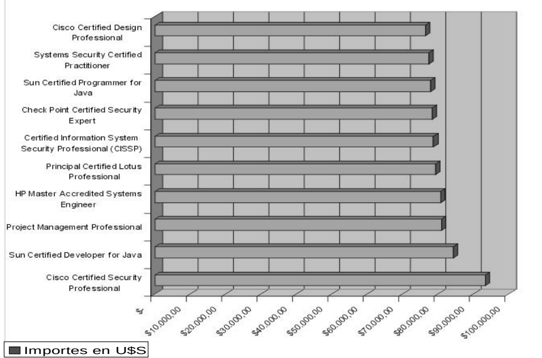
Fuente: mcp magazine



Sueldos en IT

A modo de referencia, les acercamos en esta breve reseña, 2 (dos) cuadros:
En el primero mostramos los sueldos anuales promedio, de acuerdo a las certificaciones Microsoft, en el segundo mostramos los mismos datos, de acuerdo a certificaciones no-Microsoft. Aclaramos que ambos cuadros expresan valores en dólares y corresponden a una encuesta realizada entre aprox. 6500 profesionales de IT (Information Technology) de los Estados Unidos. Aunque en Argentina no se manejen los mismos valores, estos siguen la misma tendencia.

Sueldos Promedio según otras Certificaciones



Sueldos Promedio según Certificaciones Microsoft

